

# Relatório do Uso de Gherkin:

Nomes: Thiago Medina e Mateus Gautério

Esse relatório, elaborado para a disciplina de Engenharia de Software Orientada a Modelos, proposto pelo professor Marco Mangan é baseado no exercício feito nas últimas aulas, a seguir, relataremos os passos feitos para a codificação do modelo, construção dos cenários de teste e resultados alcançados.

## 1. Modelo em Gherkin

Gherkin é uma linguagem usada para descrever comportamentos, documentação de projetos e automação de testes. Ela tem o objetivo de facilitar a comunicação entre desenvolvedores, a área de negócios e outras partes interessadas.

Basicamente, para essa atividade, elaboramos um cenário que testará nossa classe-alvo. A classe alvo é a classe “*FreqTable*” que, ao receber como entrada uma *String*, devolve uma *HashTable* que tem como formato um conjunto de caracteres e o número de vezes que cada caracter aparece na *String* de entrada.

Nosso cenário é representado da seguinte forma:

- 1- *Scenario: FreqTable sob teste*
- 2- *Given Texto de entrada ab*
- 3- *When Calcula a frequency table*
- 4- *Then Sua hashtable deve ser {b=1.0, a=1.0}*

Como pode ser observado, o primeiro cenário é um teste simples que testa a *String* de entrada *ab* e explicita que sua saída deve ser *{b=1.0, a=1.0}*.

## 2. Modelagem Adicional

Para modelagens futuras, devemos elaborar mais cenários, pensando principalmente em cenários mais detalhados, como grandes textos e também cenários que envolvam caracteres menos usuais, como símbolos e números.

### 3. Teste do Modelo

```
3 import java.util.Hashtable;
4 import org.jbehave.core.annotations.*;
5 import com.bdd.huffman.FreqTable;
6 import junit.framework.Assert;
7
8 public class HuffmanSteps {
9     private String text;
10    private FreqTable huff;
11    private double numLinhas;
12    private Hashtable<Character,Double> x;
13
14    @Given( "Texto de entrada $texto")
15    public void textoEntrada(String texto) {
16        this.text=texto;
17        huff=new FreqTable(text);
18        System.out.println("Frequence table iniciada,recebeu: "+text);
19    }
20    @When("Calcula a frequence table")
21    public void textoEntrada2() {
22        this.x= huff.getFT();
23        System.out.println("textoEntrada2 um ");
24    }
25    @Then("Sua hashtable deve ser $entrada")
26    public void resultado(String entrada) {
27        System.out.println("resultado um "+entrada);
28        System.out.println("resultado um "+x.toString());
29        Assert.assertEquals(entrada,x.toString() ); // Teste passou.
30    }
31 }
```

### 4. Endereço do Repositório

[https://github.com/MateusABG/Gherkin\\_Mateus-G-e-Thiago-M\\_Hamming](https://github.com/MateusABG/Gherkin_Mateus-G-e-Thiago-M_Hamming)

### 5. Referências

- 1- Livro Histórias de Usuários:  
[https://gallery.mailchimp.com/6f3ad17c3448b982a9c49fa6e/files/historias\\_de\\_usuario\\_3\\_0.pdf](https://gallery.mailchimp.com/6f3ad17c3448b982a9c49fa6e/files/historias_de_usuario_3_0.pdf)
- 2- Usando todo o poder de Gherkin: <https://dgosantos89.github.io/Gherkin-p1/>